

Arquitecturas Multiprocesador Distribuidas: Cluster, Grid y Cloud Computing.

Armando De Giusti², Fernando G. Tinetti¹, Marcelo Naiouf, Horacio Villagarcía¹, Franco Chichizola, Laura De Giusti, Mónica Denham, Ismael Rodríguez, Diego Montezanti³, Victoria Sanz³, Fabiana Leibovich, Emmanuel Frati³, José Pettorutti, Adrián Pousa, Enzo Rucci, Silvana Gallo, Luciano Iglesias, Javier Balladini

Instituto de Investigación en Informática LIDI (III-LIDI)
Facultad de Informática – UNLP

{degiusti, fernando, mnaiouf, hvw, francoch, ldgiusti, mdenham, ismael, dmontezanti, vsanz, fleibovich, fefrati, josep, apousa, erucci, sgallo, li}@lidi.info.unlp.edu.ar, balladini@gmail.com

Contexto

Esta línea de Investigación está dentro del proyecto “Arquitecturas Multiprocesador Distribuidas. Modelos, Software de Base y Aplicaciones” acreditado por el Ministerio de Educación y de proyectos específicos apoyados por organismos nacionales e internacionales.

En el tema hay cooperación con varias Universidades de Argentina y se está trabajando con Universidades de América Latina y Europa en proyectos financiados por CyTED, AECID y la OEI (Organización de Estados Iberoamericanos).

Se participa en iniciativas como el Programa IberoTIC de intercambio de Profesores y Alumnos de Doctorado en el área de Informática.

Por otra parte, se tiene financiamiento de Telefónica de Argentina en Becas de grado y posgrado.

Resumen

Caracterizar las arquitecturas multiprocesador distribuidas enfocadas a cluster, grid y cloud computing, con énfasis en las que utilizan procesadores de múltiples núcleos (“multicores”), con el objetivo de modelizarlas, estudiar su escalabilidad, analizar y predecir performance de aplicaciones paralelas y desarrollar esquemas de tolerancia a fallas en las mismas.

Analizar y desarrollar software de base para clusters de multicores, tratando de optimizar el rendimiento de tales arquitecturas para diferentes modelos de programación paralela y diferentes paradigmas de resolución de aplicaciones.

En el año 2011 se han agregado dos líneas de interés:

- El estudio de arquitecturas basadas en GPGPU y su comparación con clusters de multicores, así como el empleo combinado

de GPUs y multicores en computadoras de alta performance.

- El análisis de la eficiencia energética, considerando el impacto de la arquitectura, el sistema operativo, el modelo de programación y el algoritmo específico.

Es de hacer notar que este proyecto se coordina con otros dos proyectos en curso en el III-LIDI, relacionados con Algoritmos Distribuidos/Paralelos y Sistemas de Software Distribuido.

Keywords: *Sistemas Paralelos. Cluster, Multiclust, Grid y Cloud Computing. Paradigmas de programación paralela. Modelos y predicción de performance. Scheduling. Virtualización. Tolerancia a fallas. GPGPUs. Eficiencia energética.*

Introducción

La investigación en Sistemas Distribuidos y Paralelos es una de las líneas de mayor desarrollo en la Ciencia Informática actual [1][2][3]. En particular la utilización de arquitecturas multiprocesador configuradas en clusters, multiclust, grids y clouds, soportadas por redes de diferentes características y topologías se ha generalizado, tanto para el desarrollo de algoritmos paralelos, la ejecución de procesos que requieren cómputo intensivo y la atención de servicios WEB concurrentes [4][5][6][7].

El cambio tecnológico, fundamentalmente a partir de los procesadores multicore, ha impuesto la necesidad de investigar en paradigmas “híbridos”, en los cuales coexisten esquemas de memoria compartida con mensajes [8][9][10][11]. Asimismo la utilización de procesadores gráficos (GPGPUs) como arquitecturas paralelas presenta una alternativa para alcan-

¹Investigador CIC Prov. de Bs. As. ²Investigador CONICET ³Becario CONICET

zar un alto speed-up en determinadas aplicaciones [12].

Es importante en este contexto desarrollar nuevos paradigmas y herramientas para la programación eficiente de aplicaciones [13][14][15]. A su vez el concepto de *eficiencia* se refiere tanto al aspecto computacional como el energético y el impacto del consumo sobre arquitecturas con miles de procesadores que trabajan concurrentemente debe tenerse en cuenta en la estructura de los sistemas paralelos y en la planificación de la utilización de recursos por las aplicaciones [16].

Asimismo, aparecen líneas de I/D tales como el scheduling dinámico basado en el consumo del sistema paralelo y de cada subsistema (llegando al nivel de cada núcleo), el control en tiempo real de la frecuencia de reloj de los procesadores para optimizar consumo, la detección en bajo nivel de errores de concurrencia [17], el estudio y desarrollo de lenguajes, compiladores y estructuras de datos adecuados a estas arquitecturas y la detección y tolerancia a fallos tratando de minimizar el overhead de tiempo y aprovechando alguna redundancia en la misma arquitectura [14][15][18].

Por otra parte la heterogeneidad que caracteriza a los clusters y grids, así como a las redes de comunicaciones, se extiende a las nuevas arquitecturas multicore y GPGPU enfocando funcionalidades específicas para algunos núcleos, lo cual puede mejorar la performance pero al mismo tiempo complejiza el scheduling de los procesos paralelos [13][19].

La aparición de las arquitecturas tipo Cloud obliga a poner especial atención a los problemas de virtualización y predicción de performance (para la asignación dinámica de recursos). Naturalmente a mayor potencia del Cloud, también crecen las complejidades al analizar la comunicación y el acceso a memoria en arquitecturas que están distribuidas y a su vez conformadas por placas con un número variable de procesadores multicore y/o GPGPU [20][21]. En el proyecto se ha abierto una línea específicamente dedicada a los problemas de configuración y administración eficiente de Cloud, incluyendo entre los parámetros el consumo estimado de las aplicaciones [22].

Definiciones básicas

Un *procesador multicore* integra dos o más núcleos computacionales dentro de un mismo "chip" [23][24]. La motivación de su desarrollo

se basa en incrementar el rendimiento, reduciendo el consumo de energía en cada núcleo. Una GPU (Graphics Processing Unit) es una arquitectura multicore dedicada a procesamiento gráfico, con un gran número de cores simples. En los últimos años, estas arquitecturas, fueron utilizadas para aprovechar su potencia de cómputo en aplicaciones de propósito general logrando un alto rendimiento y dando lugar al concepto de GPGPU (General-Purpose Computing on Graphics Processing Units) [12][25].

Un *cluster* es un sistema de procesamiento paralelo compuesto por un conjunto de computadoras interconectadas vía algún tipo de red, las cuales cooperan configurando un recurso que se ve como "único e integrado", más allá de la distribución física de sus componentes. Cada "procesador" puede tener diferente hardware y sistema operativo, e incluso puede ser un "multiprocesador" [26]. Cuando se conectan dos o más clusters sobre una red tipo LAN o WAN, se tiene un *multicluster* [27]. La configuración más simple a considerar es la conexión de clusters homogéneos sobre una red LAN o WAN, utilizando un sistema operativo común [28].

Un *Grid* es un tipo de sistema distribuido que permite seleccionar, compartir e integrar recursos autónomos geográficamente distribuidos [28]. Un Grid es una configuración colaborativa que se puede adaptar dinámicamente según lo requerido por el usuario, la disponibilidad y potencia de cómputo de los recursos conectados. El Grid puede verse como un "entorno de procesamiento virtual", donde el usuario tiene la visión de un sistema de procesamiento "único" y en realidad trabaja con recursos dispersos geográficamente [29].

Las arquitecturas tipo "Cloud" se presentan como una evolución natural del concepto de *Clusters* y *Grids*, integrando grandes conjuntos de recursos virtuales (hardware, plataformas de desarrollo y/o servicios), fácilmente accesibles y utilizables por usuarios distribuidos, vía WEB. Estos recursos pueden ser dinámicamente reconfigurados para adaptarse a una carga variable, permitiendo optimizar su uso [21][30][31][32].

Aspectos de interés

- El incremento en el número de procesadores disponibles en clusters, grids y clouds obliga a poner énfasis en el desarrollo de los algoritmos de virtualización de modo de

- explotar la arquitectura con más de una aplicación concurrente [15].
- La heterogeneidad es inevitable en estos sistemas paralelos complejos. A su vez es un factor que condiciona la predicción de performance y consumo [19].
- A partir de la complejidad creciente del hardware, se hace más desafiante el desarrollo de capas de software eficiente, desde el middleware hasta los lenguajes de aplicación [33][34][35] [40].
- Los problemas clásicos de scheduling y mapeo de procesos a procesadores tienen nuevos objetivos (en particular los relacionados con el consumo) y deben considerar la migración dinámica de datos y procesos en función de performance y consumo [36].
- Los modelos de predicción de performance resultan especialmente complejos. Resulta de interés el estudio de esquemas sintéticos ("firmas") propios de la aplicación para estimar tiempos y consumo, ejecutando un código mínimo frente al de la aplicación real [37].
- El tema de la detección y tolerancia a fallos de hardware y software se vuelve un punto crítico al operar sobre arquitecturas con gran número de procesadores, los cuales pueden reconfigurarse dinámicamente [38][39] [41].

Líneas de Investigación y Desarrollo

Temas de Estudio e Investigación

- Arquitectura de procesadores multicore. Clusters de multicores. Software de base.
- Modelos de predicción de performance para arquitecturas tipo cluster de multicores, grids y clouds. Simulación de arquitecturas.
- Técnicas de scheduling para sistemas paralelos, en particular en función del consumo de los procesadores.
- Virtualización en clusters, grids y clouds. Predicción de performance aplicada a la virtualización.
- Nuevas estructuras de datos, orientadas a procesamiento paralelo sobre clusters, grids y clouds.
- Detección de errores de concurrencia, en tiempo de ejecución.
- Procesamiento paralelo basado en GPUs. Aplicación sobre clusters de multicores. Comparación de rendimiento con arquitecturas basadas en multicores "clásicos".
- Cloud computing. Software de base y overhead introducido por la administración de recursos en Cloud.
- Análisis comparativo de performance en cluster y cloud para problemas de HPC.
- Detección y Tolerancia a Fallos (de hardware y software) en clusters, grids y clouds.
- Métricas de evaluación de performance y escalabilidad para las nuevas arquitecturas paralelas, a partir del uso de procesadores de múltiples núcleos y/o GPGPUs..

Investigación experimental

- Desarrollo y evaluación de aplicaciones sobre cluster de multicores y multicluster heterogéneo basado en multicores. (total 192 procesadores).
- Desarrollo de software para virtualización del cluster de multicores para emular servicios de cloud computing.
- Pruebas de consumo en cluster de multicores y GPGPUs, analizando eficiencia computacional, escalabilidad y eficiencia energética.

Formación de Recursos Humanos

En cooperación con Universidades iberoamericanas se ha implementado la Maestría en Cómputo de Altas Prestaciones y se continúa dictando la Especialización en Cómputo de altas Prestaciones y Tecnología GRID. En esta línea de I/D existe cooperación a nivel nacional e internacional. Hay 8 Investigadores realizando su Doctorado, 4 realizando la Maestría y 4 alumnos avanzados están trabajando en su Tesina de Grado de Licenciatura. En 2011 se aprobó 1 Tesis Doctoral, 1 de Maestría, 3 de Especialista y 3 Tesinas de Grado en temas del proyecto.

Bibliografía

1. Grama A, Gupta A, Karypis G, Kumar V. "Introduction to parallel computing". Second Edition. Pearson Addison Wesley, 2003.
2. Dongarra J, Foster I, Fox G, Gropp W, Kennedy K, Torczon L, White A. "The Sourcebook of Parallel Computing". Morgan Kauffman Publishers. Elsevier Science, 2003.
3. Ben-Ari, M. "Principles of Concurrent and Distributed Programming, 2/E". Addison-Wesley, 2006.
4. Juhasz Z. (Editor), Kacsuk P. (Editor), Kranzmler D. (Editor). "Distributed and Parallel Systems: Cluster and Grid Computing". Springer; 1 edition (September 21, 2004).
5. Miller M. "Cloud computing: web-based applications that change the way you work and collaborate online". Que Publishing. USA 2008.

6. Di Stefano M., "Distributed data management for Grid Computing". John Wiley & Sons Inc. 2005.
7. Ghosh S. "Distributed System. An Algorithmic Approach". Chapman & Hall/CRC Computer and Information Science Series.
8. Mc. Cool M. "Programming models for scalable multicore programming". 2007. <http://www.hpcwire.com/features/17902939.html>
9. Lei Chai, Qi Gao, Dhabaleswar K. Panda. "Understanding the Impact of Multi-Core Architecture in Cluster Computing: A Case Study with Intel Dual-Core System". IEEE International Symp. on Cluster Computing and the Grid 2007 (CCGRID 2007), pp. 471-478 (May 2007).
10. Leibovich F., Gallo S., De Giusti L., Chichizola F., Naiouf M., De Giusti A. "Comparación de paradigmas de programación paralela en cluster de multicore: Pasaje de mensajes e híbrido. Un caso de estudio". Proceedings del XVII Congreso Argentino de Ciencias de la Computación (CACIC 2011), La Plata (Argentina), 2011, ISBN: 978-950-34-0756-1. Págs: 241-250.
11. Rucci E., De Giusti A., Chichizola F., Naiouf M., De Giusti L. "DNA Sequence Alignment: hybrid parallel programming on multicore cluster". Proceedings of the International Conference on Computers, Digital Communications and Computing (ICDCCC '11), Vol. 1, Nikos Mastorakis, Valeri Mladenov, Badea Lepadatescu, Hamid Reza Karimi, Costas G. Helmis (Editors), WSEAS Press, September 15-17, 2011, Barcelona, Spain, ISBN: 978-1-61804-030-5, pp. 183-190.
12. General-Purpose Computation on Graphics Processing Units. <http://gpgpu.org>.
13. De Giusti L., Chichizola F., Naiouf M., De Giusti A.E., Luque E. "Automatic Mapping Tasks to Cores - Evaluating AMTHA Algorithm in Multi-core Architectures". IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 2, No 1, March 2010. ISSN (Online): 1694-0784. ISSN (Print): 1694-0814. Págs. 1-6.
14. Olszewski M., Ansel J., Amarasinghe S. "Kendo: Efficient Deterministic Multithreading in Software". Architectural Support for Programming Languages and Operating Systems (ASPLOS '09).
15. Bertogna M., Grosclaude E., Naiouf M., De Giusti A., Luque E. "Dynamic on Demand Virtualization in High-Performance Cluster and Grid Computing. VHPC 08 – España. Agosto 2008.
16. Feng, W.C., "The importance of being low power in high-performance computing". Cyberinfrastructure Technology Watch Quarterly (CTWatch Quarterly). 2005.
17. Frati E., Olcos Herrero K., Piñuel Moreno L., Montezanti D., Naiouf M., De Giusti A. "Optimización de herramientas de monitoreo de errores de concurrencia a través de contadores de hardware". Proceedings del XVII Congreso Argentino de Ciencias de la Computación (CACIC 2011), La Plata (Argentina), 2011, ISBN: 978-950-34-0756-1. Págs: 337-346.
18. Shirako J. et al. "Compiler Control Power Saving Scheme for Multi Core Processors". LNCS Mayo 2007 – pp. 362-376.
19. Suresh Siddha, Venkatesh Pallipadi, Asit Mallick. "Process Scheduling Challenges in the Era of Multicore Processors" Intel Technology Journal, Vol. 11, Issue 04, November 2007.
20. Vaquero L.M. et al. "A Break in the Clouds: Towards a Cloud Definition". ACM SIGCOMM Computer Communication Review, vol. 39, num. 1, páginas 50-55, ISSN 0146-4833. Enero 2009.
21. Foster I. "There's Grid in them thar Clouds". 2 de Enero, 2008. <http://ianfoster.typepad.com/blog/2008/01/theres-grid-in.html>. Noviembre, 2010.
22. Rodriguez I., Pettoruti J., Chichizola F., De Giusti A. "Despliegue de un Cloud Privado para entornos de cómputo científico". Proceedings del XVII Congreso Argentino de Ciencias de la Computación (CACIC 2011), La Plata (Argentina), 2011, ISBN: 978-950-34-0756-1. Págs: 251-260.
23. Burger T. W. "Intel Multi-Core Processors: Quick Reference Guide". http://cachewww.intel.com/cd/00/00/23/19/231912_231912.pdf
24. AMD. "Evolución de la tecnología de múltiple núcleo". <http://multicore.amd.com/es-ES/AMD-Multi-Core/resources/Technology-Evolution> (2009).
25. Adrian Pousa, Victoria Sanz, Armando De Giusti, "Análisis de rendimiento de un algoritmo de criptografía simétrica sobre arquitecturas multicore", Proceedings del XVII Congreso Argentino de Ciencias de la Computación (CACIC 2011), La Plata (Argentina), 2011, ISBN: 978-950-34-0756-1. Págs: 231-240.
26. Zoltan J., Kacsuk P., Kranzlmüller D., "Distributed and Parallel Systems: Cluster and Grid Computing". The International Series in Engineering and Computer Science. Springer; 1st ed., 2004.
27. Bertogna M. L. "Planificación dinámica sobre entornos Grid". Ph.D. thesis, Universidad Nacional de La Plata, La Plata, Argentina, 2010.
28. Grid Computing and Distributed Systems (GRIDS) Laboratory - Department of Computer Science and Software Engineering (University of Melbourne). "Cluster and Grid Computing". 2007. <http://www.cs.mu.oz.au/678/>.
29. Grid Computing Infocentre: <http://www.grid-computing.com/>
30. Dikaios M. et al. "Distributed InterNet Computing for IT and Scientific Research". Internet Computing IEEE. Vol 13, Nro. 5, pp 10-13
31. Ardisson L., Goy A., Petrone G., Segnan M. "From Service Clouds to User-centric Personal Clouds". 2009 IEEE Second International Conference on Cloud Computing.
32. Hemsoth N. "Outsourcing Versus Federation: Ian Foster on Grid and Cloud". 15 de Junio, 2010. <http://www.hpcinthecloud.com/blogs/Outsourcing-Versus-Federation-Ian-Foster-on-Grid-and-Cloud-96326829.html>. Noviembre, 2010.
33. Song Y., Kalogeropoulos S., Tirumalai P. "Design and Implementation of a Compiler Framework for Helper Threading on Multi-core Processors". Pro-

- ceedings of the 14th International Conference on Parallel Architectures and Compilation Techniques; Sept. 2005.
34. Vázquez Blanco C., Huedo E., Montero R. S., Llorente I. M. "Elastic Management of Cluster-based Services in the Cloud". Proceedings pp. 19-24, ACM Digital Library 2009. ISBN 978-1-60558-564-2.
 35. Vázquez Blanco C., Huedo E., Montero R. S., Llorente I. M. "Dynamic Provision of Computing Resources from Grid Infrastructures and Cloud Providers". IEEE Society Press, pp.113-120, Workshops at the Grid and Pervasive Computing Conference, GPC 2009. ISBN 978-0-7695-3677-4.
 36. De Giusti L., Naiouf M., Chichizola F., Luque E., De Giusti A. E. "Dynamic Scheduling in Heterogeneous Multiprocessor Architectures. Efficiency Analysis". Computer Science & Technology Series – XV Argentine Congress of Computer Science Selected Papers. Editores: Guillermo Simari, Patricia Pesado, José Paganini. Págs. 85-95. ISBN 978-950-34-0684-7. Editorial de la Universidad de La Plata (edulp). La Plata (Argentina). 2010.
 37. Corredor Franco J. "Predicción de perfiles de comportamiento de aplicaciones científicas en nodos multicore". Ph.D. Thesis, Universidad Autónoma de Barcelona, Barcelona, España, Julio 2011.
 38. Lu S., Tucek J., Qin F., Zhou Y. "AVIO: detecting atomicity violations via access interleaving invariants". SIGPLAN Not., ACM, 2006, 41, 37-48.
 39. Golander A., Weiss S., Ronen R. "Synchronizing Redundant Cores in a Dynamic DMR Multicore Architecture". IEEE Transactions on Circuits and Systems II: Express Briefs Volume 56, Issue 6, 474-478. 2009.
 40. Muresano Cáceres R. "Metodología para la aplicación eficiente de aplicaciones SPMD en clústers con procesadores multicore" Ph.D. Thesis, Universidad Autónoma de Barcelona, Barcelona, España, Julio 2011.
 41. Fialho L. "Fault Tolerance configuration for uncoordinated checkpoints". Ph.D. Thesis, Universidad Autónoma de Barcelona, Barcelona, España, Julio 2011.